

```
##### Appendix #####
##### Example code for implementing the linear regression model (Table 3) and
##### for computing a confidence interval for a conditional mean estimate of
##### a Cognitive Function-Abilities theta score.
```

```
# One way of implementing the regression model
cog.for.profile <- function(thetas){
  # INPUTS: vector of scores from a PROMIS Profile Instrument
  # Note: Make sure the input thetas are in the following order:
  # depression, physical function, sleep disturbance, social roles,
  # anxiety, pain intensity.
  # OUTPUTS: estimate of the mean Cognitive Function-Abilities theta score
  # conditional on the input vector from the PROMIS Profile

  # Coefficients from Table 5 in the manuscript
  intercept <- 0.00943
  b_dep <- -0.0370
  b_phys <- 0.118
  b_slp <- -0.223
  b_social <- 0.0505
  b_anx <- -0.168
  b_pain_intensity <- -0.00599

  # Calculate estimated mean Cognitive Function-Abilities theta score
  estimate <- intercept + b_dep*thetas[1] + b_phys*thetas[2] +
    b_slp*thetas[3] + b_social*thetas[4] + b_anx*thetas[5] +
    b_pain_intensity*thetas[6]

  return(estimate)
}

# One way of estimating an alpha% confidence interval for the conditional
# expected value, in the sense of predict.lm(object, level = 1 - alpha,
# interval = "confidence")

# First, load the variance-covariance matrix of the linear model
vcov.matrix.lm <- as.matrix(read.csv("vcov-lm-updated.csv", header = FALSE),
                             mode = "numeric")

ci.for.cog <- function(thetas, alpha = 0.05){
  # INPUTS: 1) vector of scores from a PROMIS Profile Instrument
  # Note: Make sure the input thetas are in the following order:
```

```

# depression, fatigue, pain interference, physical function,
# sleep disturbance, social roles, anxiety, pain intensity
# 2) alpha level for confidence interval
# OUTPUT: a list with the following components
# - $estimate, the estimated conditional mean Cognitive Function-Abilities
# theta score
# - $lower.bound, lower bound for the alpha% confidence level for the
# conditional mean
# - $upper.bound, upper bound for the alpha% confidence level for the
# conditional mean

# Get estimated conditional mean Cognitive Function-Abilities score
estimate <- cog.for.profile(thetas)

# Augment the input vector with a 1 for the intercept
aug.theta.vector <- c(1, as.vector(thetas, mode = "numeric"))

# Compute standard error
prod <- sqrt(t(aug.theta.vector) %% vcov.matrix.lm %% aug.theta.vector)

# Get t-statistic for given alpha level
t.stat <- qt(1-alpha/2, 3973)

# Compute bounds of the confidence interval
upper.bound <- estimate + t.stat*prod
lower.bound <- estimate - t.stat*prod

# Return list with the estimate and interval bounds
return(list(estimate = estimate,
           lower.bound = lower.bound,
           upper.bound = upper.bound))
}

#### Example calculation ####

# Simulate scores for the 6 PROMIS Profile domains required for the model
# (Recall pain intensity is on a 0-10 scale)
set.seed(1)
x.sim <- runif(n = 5, min = -1, max = 1)
x.sim <- append(x.sim, sample(0:10, 1))

# Estimate conditional mean Cognitive Function-Abilities score

```

```
cog.for.profile(x.sim)
# 0.0517

# Estimate 95% confidence interval
ci.for.cog(thetas = x.sim, alpha = 0.05)
# $estimate
# 0.0517
#
# $lower.bound
# -0.0478
#
# $upper.bound
# 0.151
```